**Supervised Data Mining Project**

Name: Tharun Kumar Nagipogu

UCID: tn262

Phone: 9735838613

GitHub Link: <https://github.com/Rajtharun-nagi/DataMiningfinalProject>

**1. Objective**

The purpose of this project is to evaluate the performance of three classification algorithms on a binary classification dataset. The dataset chosen is the **Pima Indians Diabetes Dataset**, sourced from the UCI Machine Learning Repository.

The algorithms implemented include:

1. **Random Forest**: An ensemble learning method based on decision trees.
2. **Decision Tree**: A simple and interpretable classification model.
3. **LSTM (Long Short-Term Memory)**: A deep learning model for sequential data.

The project involves:

* Using **10-fold cross-validation** for model evaluation.
* Manually calculating classification metrics, including **Precision**, **Recall**, **False Positive Rate (FPR)**, **False Negative Rate (FNR)**, **F1-Score**, **True Skill Statistic (TSS)**, and **Heidke Skill Score (HSS)**.
* Comparing the performance of the models through metrics and visualizations

**2. Dataset Description**

The **Pima Indians Diabetes Dataset** contains medical diagnostic measurements to predict whether a patient has diabetes (binary outcome).

**Dataset Attributes**

* **Features**:
  + Pregnancies
  + Glucose
  + BloodPressure
  + SkinThickness
  + Insulin
  + BMI
  + DiabetesPedigreeFunction
  + Age
* **Target**:
  + Outcome (0 = non-diabetic, 1 = Diabetic)

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

**3. Methodology**

**3.1 Preprocessing**

1. **Data Loading**: The dataset was loaded using pandas and column names were added for interpretability.
2. **Feature Scaling**: Features were standardized using StandardScaler to improve model performance, especially for LSTM

**3.2 Model Implementation**

Three models were implemented:

1. **Random Forest** using scikit-learn.
2. **Decision Tree** using scikit-learn.
3. **LSTM** using TensorFlow/Keras.

Each model was evaluated using **10-fold cross-validation**. The **confusion matrix** was used to manually calculate all performance metrics for each fold.

**3.3 Metrics Calculated**

1. **Precision**: TP/TP+FP
2. **Recall**: TPTP+FN
3. **False Positive Rate (FPR)**: FPFP+TN
4. **False Negative Rate (FNR)**: FNFN+TP
5. **F1-Score**: 2⋅Precision⋅Recall / Precision+Recall
6. **True Skill Statistic (TSS)**: Recall+Specificity−1
7. **Heidke Skill Score (HSS)**: Calculated based on contingency table values.

A screenshot of a computer code

Description automatically generated

**4. Results**

**4.1 Random Forest**

**Code for Random Forest**

A screenshot of a computer program

Description automatically generated

A screen shot of a graph

Description automatically generated

**4.2 Decision Tree**

**Code for Decision TreeA screenshot of a computer program

Description automatically generated**

**A screen shot of a graph

Description automatically generated**

**4.3 LSTM**

**Code for LSTMA screenshot of a computer program

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

A graph of a curve

Description automatically generated

**4.4 ROC Curve and AUC**

The ROC curves and AUC values were plotted for each algorithm. The AUC values were:

* Random Forest: 0.88
* Decision Tree: 0.79
* LSTM: 0.91

A screenshot of a computer code

Description automatically generated

**5. Discussion**

From the results:

* **LSTM** performed best overall, with the highest Precision, Recall, and AUC.
* **Random Forest** was a close second, benefiting from ensemble learning.
* **Decision Tree** had lower performance due to its tendency to overfit.
* **6. Conclusion**
* This project successfully demonstrated the application of supervised learning algorithms on a binary classification problem, evaluated using comprehensive metrics and visualizations.

**Images for complete code comparisons:**

**A screen shot of a computer

Description automatically generated**

**A graph of different colored rectangles

Description automatically generated**

**A graph showing different colored rectangles

Description automatically generated**

**A graph of different colors

Description automatically generated with medium confidence**

**A graph with different colored squares

Description automatically generated**

**A graph showing different colored squares

Description automatically generated**

**A graph showing different colored squares

Description automatically generated**

**A graph of different colored squares

Description automatically generated**